DOI:10.1145/3418294

Tracking the historical events that lead to the interweaving of data and knowledge.

BY CLAUDIO GUTIERREZ AND JUAN F. SEQUEDA

Knowledge Graphs

"Those who cannot remember the past are condemned to repeat it."

—George Santayana

THE NOTION OF Knowledge Graph stems from scientific advancements in diverse research areas such as Semantic Web, databases, knowledge representation and reasoning, NLP, and machine learning, among others. The integration of ideas and techniques from such disparate disciplines presents a challenge to practitioners and researchers to know how current advances develop from, and are rooted in, early techniques.

Understanding the historical context and background of one's research area is of utmost importance in order to understand the possible avenues of the future. Today, this is more important than ever due to the almost infinite sea of information one faces everyday. When it comes to the Knowledge Graph area, we have noticed that students and junior researchers are not completely aware of the source of the ideas, concepts, and techniques they command.

The essential elements involved in the notion of Knowledge Graphs can be traced to ancient history in the core idea of representing knowledge in a diagrammatic form. Examples include: Aristotle and visual forms of reasoning, around 350 BC; Lull and his tree of knowledge; Linnaeus and taxonomies of the natural world; and in the 19th. century, the works on formal and diagrammatic reasoning of scientists like J.J. Sylvester, Charles Peirce and Gottlob Frege. These ideas also involve several disciplines like mathematics, philosophy, linguistics, library sciences, and psychology, among others.

This article aims to provide historical context for the roots of Knowledge Graphs grounded in the advancements of the computer science disciplines of knowledge, data, and the combination thereof, and thus, focus on the developments after the advent of computing in its modern sense (1950s). To the best of our knowledge, we are not aware of an overview of the historical roots behind the notion of knowledge graphs. We hope that this article is a contribution in this direction. This is not a survey, thus, necessarily does not cover all aspects of the phenomena and does not do a systematic qualitative or quantitative analysis of papers and systems on the topic.

This article is the authors' choice of a view of the history of the subject with

» key insights

- **Data was traditionally considered a material object, tied to bits, with no semantics per se. Knowledge was traditionally conceived as the immaterial object, living only in people's minds and language. The destinies of data and knowledge became bound together, becoming almost inseparable, by the emergence of digital computing in the mid-20th century.**
- **Knowledge Graphs can be considered the coming of age of the integration of knowledge and data at large scale with heterogeneous formats.**
- **The next generation of researchers should become aware of these developments. Both successful and not, these ideas are the basis of current technology and contain fruitful ideas to inspire future research.**

IMAGE BY ANGHI/SHUTTERSTOCK

a pedagogical emphasis directed particularly to young researchers. It presents a map and guidelines to navigate through the most relevant ideas, theories, and events that, from our perspective, have triggered current developments. The goal is to help understand what worked, what did not work, and reflect on how diverse events and results inspired future ideas.

For pedagogical considerations, we periodized the relevant ideas, techniques, and systems into five themes: Advent, Foundations, Coming-of-Age, Web Era, and Large Scale.

They follow a timeline, although with blurry boundaries. The presentation of each period is organized along two core ideas—data and knowledge—plus a discussion on data+knowledge showing their interplay. At the end of each section, we sketched a list of "realizations" (in both its senses—of becoming aware of something, as well as achievements of something desired or anticipated), and "limitations" (or, impediments) of the period. The idea is to motivate a reflection on a balance of the period. At the end of each section we include a paragraph indicating references to historical and/or technical overviews on the topics covered.

Advent of the Digital Age

The beginnings are marked by the advent and spread of digital computers and the first programming languages (LISP, FORTRAN, COBOL, and ALGOL are among the most iconic) that gave rise to the digital processing of data at massive scale and the birth of a new area of science and technology, namely, computer science. The following are five relevant threads of this era:

1. Automation of reasoning. After the first program to process complex information, "Logic Theorist" by Newell, Shaw, and Simon in 1956, they developed the "General Solving Program" in 1958, which illustrates well the paradigm researchers were after: "*this program is part of a research effort by the authors to understand the information processes that underlie human intellectual, adaptive, and creative abilities*." And the goal was stated as follows: "*to construct computer programs that can solve problems requiring intelligence and adaptation, and to discover which varieties of these programs can be matched on human problem solving*." This was continued by several other developments in the automation of reasoning, such as Robinson's Resolution Principle[33] and Green and Raphael's connection between theorem proving and deduction in databases by developing question-answering systems.[14] At the practical level there were manifold implementations of "reasoning" features. An example is Joseph Weizenbaum's

ELIZA, a program that could carry a dialogue in English on any topic, given it was programmed correctly.

2. Searching in spaces. Researchers recognized the process of searching in large spaces represented a form of "intelligence" or "reasoning." Having an understanding of such space would ease searching. Sorting is a simple example. Easily 25% of computer time until the 1970s was used in sorting data to make feasible any search procedure.[6] The very notion of search was well known to people working in data processing, even before the advent of computers. However, the idea of searching in diverse and complex spaces was new, such as search spaces arising in games (for example, chess, checkers, and Go). Dijkstra's famous algorithm for finding shortests paths is from 1956, and its generalization A* is from 1968.[19]

3. Retrieving information from unstructured sources. Once having the computational capabilities, one can get data from sources beyond traditional structured data. The ideas go back to V. Bush's report "As We May Think" but were developed systematically in the 1950s.[11] A milestone was Bertram Raphael's "SIR: A Computer Program for Semantic Information Retrieval" (1964).[31] This system demonstrated what could be called an ability to "understand" semantic information. It uses word associations and property lists for the relational information normally conveyed in conversational statements. A format-matching procedure extracts semantic content from English sentences.

4. Languages and systems to manage data. An early system to manage data was the Integrated Data Store (IDS) designed by Charles Bachman in 1963.[2] The IDS system maintained a collection of shared files on disk, had tools to structure and maintain them, and an application language to manipulate data. This allowed efficiency at the cost of what was later called "data independence." IDS became the basis for the CODASYL standard, which became known as Database Management Systems (DBMS). Furthermore, the idea that there should be more dedicated languages to handle data led to the creation of COBOL (1959), which is an early example of a programming language oriented to data handling and with a syntax resembling English.

5. Graphical representation of knowledge. Semantic networks were introduced in 1956 by Richard H. Richens, a botanist and computational linguist, as a tool in the area of machine translation of natural languages.[32] The notion was developed independently by several people. Ross Quillian's 1963 paper "A Notation for Representing Conceptual Information: An Application to Semantics and Mechanical English Paraphrasing" aimed at allowing information "to be stored and processed in a computer" following the model of human memory. The idea of searching for "design principles for a large memory that can enable it to serve as the base of knowledge underlying human-like language behavior" was further developed in his doctoral dissertation "Word concepts: A theory and simulation of some basic semantic capabilities" in 1967.[29]

Sketch of realizations and limitations in the period. Among the realizations, the following stand out: The awareness of the importance and possibilities of automated reasoning; the problem of dealing with large search spaces; the need to understand natural language and other human representations of knowledge; the potential of semantic nets (and graphical representations in general) as abstraction layers; and the relevance of systems and high level languages to manage data. Regarding limitations, among the most salient were: the limited capabilities (physical and technical) of hardware; the availability and high cost of hardware; the gap between graphical representation and sequential implementation; and the gap between the logic of human language and the handling of data by computer systems.

Overview and secondary sources. For computing, P.E. Ceruzzi, *History of Modern Computing*; for the history of AI: N.J. Nilsson, *The Quest for Artificial Intelligence*.

Data and Knowledge Foundations

The 1970s witnessed much wider adoption of computing in industry. These are the years when companies such as Apple and Microsoft were founded. Data processing systems such as Wordstar and VisiCalc, predecessors of current personal word processors and spreadsheets, were born. The increasing storage and processing power, as well as human expertise drove the need to improve how data should be managed for large companies.

Data. The growth in data processing needs brought a division of labor expressed in the notion of *representational independence*. Programmers and applications could now "forget" how the data was physically structured in order to access data. This idea is at the core of Edgar Codd's paper "A Relational Model of Data for Large Shared Data Banks"[8] that describes the use of relations as a mathematical model to provide representational independence; Codd calls this "data independence." This theory and design philosophy fostered database management systems and modeling tools.

At the modeling level, Peter Chen introduced a graphical data model in his paper "The Entity-Relationship Model: Toward a Unified View of Data,"[7] which advocated modeling data based on entities and relationships between them. Such ER models incorporated semantic information of the real world in the form of graphs. It is one of the early attempts to link a conceptual design with a data model—in this case the relational data model.

At the system level, software applications were developed and implemented to manage data based on the relational model, known as Relational Database Management Systems (RDBMS). Two key systems during this time were IBM's System R, described in the paper "System R: Relational Approach to Database Management" (1976), and University of California at Berkeley's INGRES, described in "The Design and Implementation of INGRES" (1976). These systems were the first to implement the "vision" of the relational model as described by Codd, including relational query languages such as SEQUEL and QUEL, which would lead to SQL, the most successful declarative query language in existence.

Knowledge. While the data stream was focusing on the structure of data and creating systems to best manage it, knowledge was focusing on the meaning of data. An early development in this direction was the work of

S.C. Shapiro who proposed a network data structure for organizing and retrieving semantic information.[34] These ideas were implemented in the semantic network and processing system (SNePS) that can be considered as one of the first stand-alone KRR systems.

In the mid-1970s, several critiques to semantic network structures emerged, focusing on their weak logical foundation. A representation of this criticism was William Woods' 1975 paper "What's in a Link: Foundations for Semantic Networks."[40]

Researchers focused on extending semantic networks with formal semantics. An early approach to providing structure and extensibility to local and minute knowledge was the notion of *frames*. This was introduced by Marvin Minsky in his 1974 article "A Framework for Representing Knowledge."[27] A frame was defined as a network of nodes and relations. In 1976, John Sowa introduced Conceptual Graphs in his paper "Conceptual Graphs for a Data Base Interface."[36] Conceptual graphs serve as an intermediate language to map natural language queries and assertions to a relational database. The formalism represented a sorted logic with types for concepts and relations. In his 1977 paper "In Defense of Logic," Patrick Hayes recognized that frame networks could be formalized using first order logic.[20] This work would later influence Brachman and Levesque to identify a tractable subset of First-order logic, which would become the first development in Description Logics (see next section).

Data + Knowledge. In the 1970s, data and knowledge started to experience an integration. Robert Kowalski, in "Predicate Logic as Programming Language,"[23] introduced the use of logic as both a declarative and procedural representation of knowledge, a field now known as logic programming. These ideas were implemented by Alain Colmerauer in PROLOG.

Early systems that could reason based on knowledge, known as knowledge-based systems, and solve complex problems were expert systems. These systems encoded domain knowledge as if-then rules. R. Davis, B. Buchanan, and E. Shortliffe were among the first to develop a successful expert system, MYCIN, that became a classic example to select antibiotic therapy for bacteremia.[10] An open problem was understanding where to obtain the data and knowledge. This area would be called knowledge acquisition.

Conceptual graphs serve as an intermediate language to map natural language queries and assertions to a relational database.

The 1977 workshop on "Logic and Data Bases," held in Toulouse, France, and organized by Herve Gallaire, Jack Minker, and Jean-Marie Nicolas,[13] is considered a landmark event. Important notions such as Closed World Assumption by Ray Reiter and Negation as Failure by Keith Clark were presented at this workshop, which can be considered the birth of the logical approach to data. Many researchers consider this to be the event that formalized the link between logic and databases, designating it as a field on its own.

Sketch of realizations and limitations in the period. Realizations of this period include: the need for and potential of representational independence, as shown by the case of the relational model; practical and successful implementations of the relational model; the realization that semantic networks require formal frameworks using the tools of formal logic; and the awareness of the potential of combining logic and data by means of networks.The limitations include: on the data side, the inflexibility of traditional data structures to represent new varieties of data (which gave rise to object-oriented and graph data structures); on the knowledge side, weakness of the logical formalization of common knowledge (which will be the motive of the rise of description logics).

Overview and secondary sources. On logic programming: A. Colmerauer and Ph. Roussel, *The Birth of Prolog*; R. Kowalski, *The Early Years of Logic Programming*. On knowledge representation: R.H. Brachman, H.J. Levesque, *Readings in Knowledge Representation*. On Expert Systems: F. Puppe, *Systematic introduction to Expert Systems*, Ch.1.

Coming-of-Age of Data and Knowledge

The 1980s saw the evolution of computing as it transitioned from industry to homes through the boom of personal computers. In the field of data management, the Relational Database industry was developing rapidly (Oracle, Sybase, IBM, among others). Object-oriented abstractions

were developed as a new form of representational independence. The Internet changed the way people communicated and exchanged information.

Data. Increasing computational power pushed the development of new computing fields and artifacts. These, in turn, generated complex data that needed to be managed. Furthermore, the relational revolution, which postulated the need of representational independence led to a separation of the software program from the data. This drove the need to find ways to combine object-oriented programming languages with databases. This gave rise to the development of object-oriented databases (OODB). This area of research investigated how to handle complex data by incorporating features that would become central in the future of data, such as objects, identifiers, relationships, inheritance, equality, and so on. Many systems from academia and industry flourished during this time, such as Encore-Observer (Brown University), EXODUS (University of Wisconsin–Madison), IRIS (Hewlett-Packard), ODE (Bell Labs), ORION (MCC), and Zeitgeist (Texas Instruments), which led to several commercial offerings.

Graphs started to be investigated as a representation for object-oriented data, graphical and visual interfaces, hypertext, etc. An early case was Harel's higraphs,[18] which formalize relations in a visual structure, and are now widely used in UML. Alberto Mendelzon and his students developed the early graph query languages using recursion.[9] This work would become the basis of modern graph query languages.

Knowledge. An important achievement in the 1980s was understanding the trade-off between the expressive power of a logic language and the computational complexity of reasoning tasks. Brachman and Levesque's paper "The Tractability of Subsumption in Frame-Based Description Languages" was among the first to highlight this issue.[4] By increasing the expressive power in a logic language, the computational complexity increases. This led to research trade-offs along the expressivity continuum, giving rise to a new family of logics called *Description Logics*. Standout systems are

Increasing computational power pushed the development of new computing fields and artifacts. These, in turn, generated complex data that needed to be managed.

KL-ONE, LOOM, and CLASSIC, among others. In addition to Description Logic, another formalism was also being developed at that time: F-Logic was heavily influenced by objects and frames, allowing it to reason about schema and object structures within the same declarative language.[22]

These early logic systems showed that logical reasoning could be implemented in tractable software. They would become the underpinning to OWL, the ontology language for the Semantic Web.

Additionally, the development of non-monotonic reasoning techniques occurred during this time, for example, the introduction of numerous formalisms for non-monotonic reasoning, including circumscription, default logic, autoepistemic logics and conditional logics.

Data + Knowledge. A relevant development in the 1980s was the Japanese 5th Generation Project.

Given Japan's success in the automotive and electronics industries, they were looking to succeed in software. The goal was to create artificial intelligence hardware and software that would combine logic and data and could carry on conversations, translate languages, interpret pictures, and reason like human beings. The Japanese adopted logic programming as a basis to combine logic and data.

The Japanese project sparked world wide activity leading to competing projects such as Microelectronics and Computer Technology Consortium (MCC) in the U.S., the European Computer Research Centre (ECRC) in Munich, and the Alvey Project in the U.K. MCC was an important research hub, both in hardware and software throughout the 1980s and 1990s. For example, the Cyc project, which came out of MCC, had the goal of creating the world's largest knowledge base of common sense to be used for applications performing human-like reasoning.

Expert systems proliferated in the 1980s and were at the center of the AI hype. We see the development of production rule systems such as OPS5, the Rete algorithm,[12] and Treat algorithm to efficiently implement rule-based systems. Expert systems were deployed on parallel computers, such as the DADO Parallel Computer, the Connection

Machine, and the PARKA Project, among others. Expert systems started to show business value (for example, Xcon, ACE). Venture capitalists started to invest in AI companies such as IntelliCorp, ILOG, Neuron Data, and Haley Systems, among others.

On the academic side, an initial approach of combining logic and data was to layer logic programming on top of relational databases. Given that logic programs specify functionality ("the what") without specifying an algorithm ("the how"), optimization plays a key role and was considered much harder than the relational query optimization problem. This gave rise to deductive databases systems, which natively extended relational databases with recursive rules. Datalog, a subset of Prolog for relational data with a clean semantics, became the query language for deductive databases.[5] One of the first deductive databases systems was the LDL system, presented in Tsur and Zaniolo's paper "LDL: A Logic-Based Data-Language."[37] Many of these ideas were manifested directly in relational databases known then as active databases.

At the beginning of the 1990s, expert systems proved expensive and difficult to update and maintain. It was hard to explain deductions, they were brittle, and limited to specific domains. Thus the IT world moved on and rolled that experience into mainstream IT tools from vendors such as IBM, SAP, andOracle, among others. A decade after the start of the Japanese 5th Generation project, its original impressive list of goals had not been met. Funding dried out and these factors led to what has been called an AI Winter.

By the end of this decade, the first systematic study with the term "Knowledge Graph" appeared. It was the Ph.D. thesis of R.R. Bakker, "Knowledge Graphs: Representation and Structuring of Scientific Knowledge." Many of these ideas were published later (1991) in a report authored by P. James (a name representing many researchers) and titled "Knowledge Graphs."[21] The term did not permeate widely until the second decade of the next century.

Sketch of realizations and limitations in the period. Among the most important realizations were the fact that the integration between logic and data must be tightly coupled—that is, it is not enough to layer Prolog/expert systems on top of a database; and the relevance of the trade-off between expressive power of logical languages and the computational complexity of reasoning tasks. Two main limitations deserve to be highlighted: the fact that negation was a hard problem and was still not well understood at this time; and that reasoning at large scale was an insurmountable problem—in particular, hardware was not ready for the task. This would be known as the knowledge acquisition bottleneck.

Overview and secondary sources. On the golden years of graph databases, see R. Angles, C. Gutierrez, *Survey of Graph Database Models*. On O-O databases: M. Atkinson et al., T*he Object-Oriented Database System Manifesto*. On the Japanese 5th Generation Project: E. Shapiro et.al. *The 5th Generation Project: Personal Perspectives*.

Data, Knowledge, and the Web

The 1990s witnessed two phenomena that would change the world. First, the emergence of the World Wide Web, the global information infrastructure that revolutionized traditional data, information, and knowledge practices. The idea of a universal space of information where anybody could post and read, starting with text and images, in a distributed manner, changed completely the philosophy and practices of knowledge and data management. Second, the digitization of almost all aspects of our society. Everything started to move from paper to electronic. These phenomena paved the way to what is known today as Big Data. Both research and industry moved to these new areas of development.

Data. The database industry focused on developing and tuning RDBMS to address the demands posed by e-commerce popularized via the Web. This led to the generation of large amounts of data which were required to be integrated and analyzed. Research built on this momentum and focused on the areas of web data, data integration, data warehouse/OLAP, and data mining.

The data community moved toward the Web. Diverse efforts helped in developing an understanding of data and computations on the Web, shown in papers such as "Formal Models of the Web" by Mendelzon and Milo[26] and "Queries and Computation on the Web" by Abiteboul and Vianu.[1] The Web triggered the need for distributing self-describing data. A key result of fulfilling these goals was semistructured data models, such as Object Exchange Model (OEM), Extensible Markup Language (XML), and Resource Description Framework (RDF), among others.

During this time, organizations required integration of multiple, distributed, and heterogeneous data sources in order to make business decisions. Federated databases had started to address this problem in the 1980s (see survey[35]). During this period, industry and academia joined forces and developed projects such as TSIMMIS and Lore from Stanford/IBM, SIMS from USC, InfoSleuth from MCC, among many others. These systems introduced the notion of mediators and wrappers.[39] Systems such as SIMS and InfoSleuth also introduced ontologies into the data integration mix.

In this context, due to the amount of data being generated and integrated, there was a need to drive business decision reporting. This gave rise to data warehouse systems with data modeled in star and snowflake schemas. These systems could support analytics on multi-dimensional data cubes, known as on-line analytical processing (OLAP). Much of the research focused on coming up with heuristics to implement query optimizations for data cubes. Business needs drove the development of data mining techniques to discover patterns in data.

Knowledge. Researchers realized that knowledge acquisition was the bottleneck to implement knowledge-based and expert systems. The Knowledge Acquisition Workshops (KAW in Canada and EKAW in Europe) were a series of events where researchers discussed the knowledge acquisition bottleneck problem. The topic evolved and grew into the fields of knowledge engineering and ontology engineering.

The Web was a realization that knowledge, not just data, should also be shared and reused. The need to elevate from administrative metadata to formal semantic descriptions gave rise to the spread of languages

to describe and reason over taxonomies and ontologies.

The notion of ontology was defined as a "shared and formal specification of a conceptualization" by Gruber.[15]

Among the first scientists arguing the relevance of ontologies were N. Guarino,[16] M. Uschold, and M. Grunninger.[38] Research focused on methodologies to design and maintain ontologies, such as METHONOLOGY, Knowledge Acquisition and Documentation Structuring (KADS) methodology, CommonKADS, and specialized methods such as OntoClean. We observe the emergence of the first ontology engineering tools (for example, Ontolingua, WebODE, and Protege) to help users code knowledge.

Data + Knowledge. The combination of data and knowledge in database management systems was manifested through *Deductive Databases*. Specialized workshops on Deductive Databases (1990–1999) and Knowledge Representation meets Databases (1994–2003) were a center for the activity of the field.[30] These developments led to refined versions of Datalog, such as probabilistic, disjunctive, and Datalog +/-.

An important challenge driving research was how to cope with formal reasoning at Web scale. In fact, viewing the Web as a universal space of data and knowledge, drove the need to develop languages for describing, querying and reasoning over this vast universe. The Semantic Web project is an endeavor to combine knowledge and data on the Web. The following developments influenced and framed the Semantic Web project: Simple HTML Ontology Extensions (SHOE), Ontobroker, Ontology Inference Layer (OIL) and DARPA Agent Markup Language (DAML), Knowledge Query and Manipulation Language (KQML), and the EU-funded Thematic Network OntoWeb (ontology-based information exchange for knowledge management and e-commerce) among others. The goal was to converge technologies such as knowledge representation, ontologies, logic, databases, and information retrieval on the Web. These developments gave rise to a new field of research and practice centered around the Web and its possibilities.

Sketch of realizations and limitations in the period. The main realization was that the Web was rapidly starting to change the ways the world of data, information and knowledge was traditionally conceived; new types of data were proliferating, particularly media data like images, video, and voice; and finally, the awareness that data must be—and now can be—connected to get value. Among the limitations is worth mentioning that the computational power was not enough to handle the new levels of data produced by the Web; and that the pure logical techniques have complexity bounds that made their scalability to certain growing areas like searching and pattern matching very difficult and at times infeasible.

Overview and secondary sources. About the Web: T. Berners-Lee, *Weaving the Web*. On data and the Web: S. Abiteboul et al., *Data on the Web: From Relations to Semistructured Data and XML*. On Ontology Engineering: R. Studer et al., *Knowledge Engineering: Principles and Methods*. On Web Ontology Languages: I. Horrocks et al., *From SHIQ and RDF to OWL: The Making of a Web Ontology Language*.

Data and Knowledge at Large Scale

The 2000s saw the explosion of e-commerce and online social networks (Facebook, Twitter, and so on). Advances in hardware and new systems made it possible to generate, store, process, manage, and analyze data at a much larger scale. We entered the Big Data revolution. During this era, we see the rise of statistical methods by the introduction of deep learning into AI.

Data. Web companies such as Google and Amazon pushed the barrier on data management.

Google introduced an infrastructure to process large amounts of data with MapReduce. The emergence of non-relational, distributed, data stores got a boom with systems such as CouchDB, Google Bigtable and Amazon Dynamo. This gave rise to "NoSQL" databases that (re-)popularized database management systems for Column, Document, Key-Value and Graph data models.

Many of the developments were triggered by the feasibility to handle and process formats like text, sound, images, and video. Speech and image recognition, image social networks like Flickr, advances in NLP, and so on consolidated the notion that "data" is well beyond tables of values.

The data management research community continued its research on data integration problems such as schema matching, entity linking, and XML processing. Database theory researchers studied data integration and data exchange from a foundational point of view.[25]

Knowledge. The Description Logic research community continued to study trade-offs and define new profiles of logic for knowledge representation. Reasoning algorithms were implemented in software systems (for example, FACT, Hermit, Pellet). The results materialized as the European Ontology Inference Layer (OIL) DARPA Agent Markup Language (DAML) infrastructure. Both efforts joined forces and generated DAML+OIL, a thin ontology layer built on RDF with formal semantics based on description logics. This influenced the standardization of the Web Ontology Language (OWL) in 2004, which is a basis for the Semantic Web.

Big Data drove statistical applications to knowledge via machine learning and neural networks. Statistical techniques advanced applications that deduced new facts from already known facts. The 2012 work on image classification with deep convolutional neural networks with GPUs[24] is signaled as a result that initiated a new phase in AI: deep learning.

The original attempts in the 1960s to model knowledge directly through neural networks were working in practice. These techniques and systems now would outperform many human specific tasks such as classification, and applications where large amounts of training data and powerful hardware are available.

Data + Knowledge. The connection between data and knowledge was developed in this period along two lines, namely logical and statistical.

On the logical thread, the Semantic Web project was established, built upon previous results like the graph data model, description logics, and knowledge engineering.

The paper "The Semantic Web" by Tim Berners-Lee, Jim Hendler and Ora

Lassila[3] sparked an excitement from industry and academia. The technologies underpinning the Semantic Web were being developed simultaneously by academia and industry through the World Wide Web Consortium (W3C) standardization efforts. These resulted in Resource Description Framework (RDF), Web Ontology Language (OWL), and SPARQL Protocol and RDF Query Language (SPARQL), among others.

In 2006, Tim Berners-Lee coined the term "Linked Data" to design a set of best practices highlighting the network structure of data on the Web in order to enhance knowledge.

This gave rise to the Linked Open Data (LOD) project and large RDF graph-based knowledge bases such as DBPedia, and Freebase, which would eventually lead to Wikidata. The LOD project was a demonstration of how data could be integrated at Web scale. In 2011, the major search engines released schema.org, a lightweight ontology, as a way to improve the semantic annotation of Web pages. These efforts were built on the results of the Semantic Web research community.

On the statistical thread, the beginning of the 21st century witnessed advances and successes in statistical techniques for large-scale data processing such as speech recognition, NLP, and image processing. This motivated Halevy, Norvig, and Pereira to speak of the "the unreasonable effectiveness of data."[17] This is probably one of the drivers that motivated the search for new forms of storing, managing and integrating data and knowledge in the world of Big Data and the emergence of the notion of Knowledge Graph. Furthermore, researchers have been making efforts to address statistical phenomena while incorporating techniques from logic and traditional databases such as statistical relational learning since the 1990s. Finally, it is relevant to highlight a new field dealing with data and knowledge that emerged under these influences: Data science.

Sketch of realizations and limitations in the period. Among the realizations in this period, we learned to think about data and knowledge in a much bigger way, namely at Web scale; and the world of data entered the era of neural networks due to new hardware and clever learning techniques. One of the main limitations that made advances in this area difficult, is the fact that, although people realized the need to combine logical and statistical techniques, little is yet known on how to integrate these approaches. Another important limitation is that statistical methods, particularly in neural networks, still are opaque regarding explanation of their results.

The beginning of the 21st century witnessed advances and successes in statistical techniques for large-scale data processing such as speech recognition, NLP, and image processing.

Overview and secondary sources. D. Agrawal et al., *Challenges and Opportunities with Big Data*. T. Hey et al. *The Fourth Paradigm: Data-Intensive Scientific Discovery*. R. Fagin et al. *Reasoning About Knowledge*.

Where Are We Now?

A noticeable phenomenon in the history we have sketched is the never-ending growth of data and knowledge, in both size and diversity. At the same time, an enormous diversity of ideas, theories, and techniques were being developed to deal with it. Sometimes they reached success and sometimes ended in failure, depending on physical and social constraints whose parameters most of the time were far out of the researcher's control.

In this framework, historical accounts can be seen as a reminder that absolute success or failure does not exist, and that each idea, theory, or technique needs the right circumstances to develop its full potential. This is the case with the notion of Knowledge Graphs. In 2012, Google announced a product called the Google Knowledge Graph. Old ideas achieved worldwide popularity as technical limitations were overcome and it was adopted by large companies. In parallel, other types of "Graph" services were developed, as witnessed by similar ideas by other giants like Microsoft, Facebook, Amazon and Ebay.[28] Later, myriad companies and organizations started to use the Knowledge Graph keyword to refer to the integration of data, given rise to entities and relations forming graphs. Academia began to adopt this keyword to loosely designate systems that integrate data with some structure of graphs, a reincarnation of the Semantic Web, and Linked Data. In fact, today the notion of Knowledge Graph can be considered, more than a precise notion or system, an evolving project and a vision.

The ongoing area of Knowledge Graphs represents in this sense a convergence of data and knowledge techniques around the old notion of graphs or networks. From the data tradition, database technologies, and systems began to be developed by various companies and academia; manifold graph query languages are being developed: standard languages such as SPARQL and SPARQL 1.1, new industrial languages like Cypher, GSQL, and PGQL, research languages such as G-CORE, and the upcoming ISO standard GQL. On the other hand, we see a wealth of knowledge technologies addressing the graph model: on the logical side, the materialization and implementation of old ideas like semantic networks, and frames, or more recently, the Semantic Web and Linked Data projects; on the statistical side, techniques to extract, learn, and code knowledge from data on a large scale through knowledge graph embeddings.

It is not easy to predict the future, particularly the outcome of the interplay between data and knowledge, between statistics and logic. Today we are seeing a convergence of statistical and logical methods, with the former temporarily overshadowing the latter in the public eye. It is for this reason that we consider it relevant to call attention to history and "recover" the long-term significance of the achievements in the areas of data and knowledge. As we pointed out, even though some ideas and developments of the past may not have been successful or well known (or even known at all) at the time, they surely contain fruitful ideas to inspire and guide future research.

If we were to summarize in one paragraph the essence of the developments of the half century we have presented, it would be the following: Data was traditionally considered a commodity, moreover, a material commodity—something given, with no semantics per se, tied to formats, bits, matter. Knowledge traditionally was conceived as the paradigmatic "immaterial" object, living only in people's minds and language. We have tried to show that since the second half of the 20th century, the destinies of data and knowledge became bound together by computing.

We have attempted to document how generations of computing scientists have developed ideas, techniques, and systems to provide material support for knowledge and to elevate data to the conceptual place it deserves.

Acknowledgments

This work was funded by ANID – Millennium Science Initiative Program – Code ICN17_002.

We reached out to many colleagues asking for their input on this article. We are extremely thankful for their helpful feedback: Harold Boley, Isabel Cruz, Jerome Euzenat, Dieter Fensel, Tim Finin, Enrico Franconi, Yolanda Gil, Joe Hellerstein, Jim Hendler, Jan Hidders, Ian Horrocks, Bob Kowalski, Georg Lausen, Leonid Libkin, Enrico Motta, Misty Nodine, Natasha Noy, Amit Sheth, Steffen Staab, Rudi Studer, Michael Uschold, Frank van Harmelen, Victor Vianu, Darrell Woelk, and Peter Wood. Juan thanks Daniel Miranker for inspiration on the topic of this article. We also thank Schloss Dagstuhl for hosting us in 2017 and 2018 to do this research and copyeditor Melinda O'Connell.

References

1. Abiteboul, S. and Vianu, V. Queries and computation on the Web. In *Proceedings of the 6th Intern. Conf. Database Theory*, 1997.
2. Bachman, C.W. The origin of the integrated data store (IDS): The first direct-access DBMS. *IEEE Ann. Hist. Comput. 31*, 4 (Oct. 2009), 42–54.
3. Berners-Lee, T., James Hendler, J. and Ora Lassila, O. The Semantic Web. *Sci. Amer. 5* (May 2001), 34–43.
4. Brachman, R.J and Levesque, H.J. The tractability of subsumption in frame-based description languages. In *Proceedings of the Nat. Conf. Artificial Intelligence*. (Austin, TX, USA, Aug. 6–10, 1984), 34–37.
5. Ceri, S., Gottlob, G., and Tanca, L. What you always wanted to know about datalog (and never dared to ask). *IEEE Trans. Knowl. Data Eng. 1*, 1 (1989), 146–166.
6. Ceruzzi, P.E. *A History of Modern Computing* (2 ed.). MIT Press, Cambridge, MA, USA, 2003.
7. Chen, P.P. The entity-relationship model—Toward a unified view of data. *ACM Trans. Database Syst. 1*, 1 (1976), 9–36.
8. Codd, E.F. A relational model of data for large shared data banks. *Commun. ACM 13*, 6 (1970), 377–387.
9. Cruz, I.F, Mendelzon, A.O. and Wood, P.T A graphical query language supporting recursion. *SIGMOD*, 1987, 323–330.
10. Davis, R., Buchanan, B., and Shortliffe, E. Production rules as a representation for a knowledge-based consultation program. *Artif. Intell. 8*, 1 (Feb. 1977), 15–45.
11. Fairthorne, R.A. Automatic retrieval of recorded information. Comput. J. 1, 1 (Jan. 1958), 36–41.
12. Forgy, C. Rete: A fast algorithm for the many patterns/many objects match problem. *Artif. Intell. 19*, 1 (1982), 17–37.
13. Gallaire, H. and Minker, J. (Eds.). *Proceedings of the Symposium on Logic and Data Bases*, Centre d'études et de recherches de Toulouse, France, 1977.
14. Green, C.C. and Raphael, B. The use of theorem-proving techniques in question-answering systems. In *Proceedings of the 1968 23rd ACM National Conf.*, 169–181.
15. Gruber, T.R. Toward principles for the design of ontologies used for knowledge sharing. *Int. J. Hum.-Comput. Stud. 43*, 5-6 (Dec. 1995), 907–928.
16. Guarino, N. Formal ontology, conceptual analysis and knowledge representation. *Int. J. Hum.-Comput. Stud. 43*, 5-6 (Dec. 1995), 625–640.
17. Halevy, A.Y., Norvig, P. and Pereira, F. The unreasonable effectiveness of data. *IEEE Intell. Syst. 24*, 2 (2009), 8–12.
18. Harel, D. On visual formalisms. *Commun. ACM 31*, 5 (1988), 514–530.
19. Hart, P.E., Nilsson, N.J., and Raphael, B. A formal basis for the heuristic determination of minimum cost paths. *IEEE Trans. Systems Science and Cybernetics 4*, 2 (1968), 100–107.
20. Patrick J. Hayes. 1977. In Defense of Logic. In *IJCAI*. 559–565.
21. James, P. Knowledge graphs. Number 945 in Memorandum Faculty of Applied Mathematics. University of Twente, Faculty of Applied Mathematics, 1991.
22. Kifer, M., Lausen, G., and Wu, J. Logical foundations of object-oriented and frame-based languages. *J. ACM 42*, 4 (1995), 741–843.
23. Kowalski, R.A. Predicate logic as programming language. In *Proceedings of the 6th IFIP Congress on Information Processing*, 1974, 569–574.
24. Krizhevsky, A., Sutskever, I. and Hinton, G.E. ImageNet classification with deep convolutional neural networks. In *Proceedings of NIPS*.
25. Lenzerini, M. Data integration: A theoretical perspective. In *Proceedings of PODS '02*, 233–246.
26. Mendelzon, A.O. and Milo, T. Formal models of Web queries. In *Proceedings of PODS '97*, 134–143.
27. Minsky, M. A Framework for Representing Knowledge. Technical Report, 1974, Cambridge, MA, USA.
28. Noy, N.F., Gao, Y., Jain, A., Narayanan, A., Patterson, A., and Taylor, J. Industry-scale knowledge graphs: lessons and challenges. *Commun. ACM 62*, 8 (Aug. 2019), 36–43.
29. Quillian, R.M. Word concepts: A theory and simulation of some basic semantic capabilities. *Behavioral Science 12* (1967), 410–430.
30. Ramakrishnan, R. and Ullman, J.D. A survey of deductive database systems. *J. Log. Program. 23*, 2 (1995), 125–149.
31. B. Raphael, B. SIR: A Computer Program for Semantic Information Retrieval. Technical Report, 1964, Cambridge, MA, USA.
32. Richens, R.H. Preprogramming for mechanical translation. *Mechanical Translation 3*, 1 (1956), 20–25.
33. Robinson, J.A. A machine-oriented logic based on the resolution principle. *J. ACM 12*, 1 (1965), 23–41.
34. Shapiro, S.C. 1971. A net structure for semantic information storage, deduction and retrieval. In *Proceedings of the 2nd Intern. Joint Conf. Artificial Intelligence*. (London, U.K., Sept. 1-3, 1971), 512–523.
35. Sheth, A.P. and Larson, J.A. Federated database systems for managing distributed, heterogeneous, and autonomous databases. *ACM Comput. Surv. 22*, 3 (Sept. 1990), 183–236.
36. Sowa, J.F. Conceptual graphs for a data base Iinterface. *IBM J. Research and Development 20*, 4 (1976), 336–357.
37. Tsur, S. and Zaniolo, C. LDL: A logic-based data language. In *Proceedings of the 12th Intern. Conf. on Very Large Data Bases*, 1986, 33–41.
38. Uschold, M. and Gruninger, M. Ontologies: Principles, methods and applications. *Knowledge Eng. Review 11*, 2 (1996), 93–136.
39. Wiederhold, G. Mediation in information systems. *ACM Comput. Surv. 27*, 2 (June 1995), 265–267.
40. Woods, W. What's in a link: Foundations for semantic networks. 76 (Nov. 1975); https://doi.org/10.1016/B978- 1- 4832- 1446- 7.50014- 5.

For an extended collection of references and resources, see the online appendix at https://dl.acm.org/doi/10.1145/3418294

Claudio Gutierrez (cgutierr@dcc.uchile.cl) is a professor at the DCC, Universidad de Chile and IMFD.

Juan F. Sequeda (juan@data.world) is a principal scientist at data.world, Austin, TX, USA.